ality to the point P_1 actually touched by a pen. Assuming that at the point P_1 an input actuator is set onto the touch pad before an input on one of the input areas **102** is expected. When P_1 and the input areas **102** are known and the only performable dual-point input comprises an input to one of the input areas **102**, the boundary areas **104** can be calculated. Dual-point input is then only enabled if and when a discontinuous jump into one of the boundary areas **104** is detected. If a movement of point P_M is detected, the point P_2 within the input areas **102** are used as reference points to calculate the movements of P_1 from the movements of P_M.

[**0106**] In the matrix number **4** the number of possible dual point inputs are considerably reduced as compared with the conventional methods. The boundary areas **104** can be regarded as a kind of input prediction used to increase recognition accuracy of dual-point inputs. It may be noted that the matrix **4** is embodied as a matrix for left handed users wherein the input areas **102** are operated by e.g. the thumb of the right hand, and therefore are located at the right side of the matrix **4**.

[**0107**] **FIG. 10** is a flow chart of another implementation of the method of the present invention. Basically the method comprises the same steps as disclosed in **FIG. 8**, and therefore the similar steps are not described, but reference is made to the description of **FIG. 8**.

[**0108**] The method differs from the one disclosed **FIG. 8** by an additional inquiry step **11** inserted after the detection **80** of an input event at the point P_1, to determine if the input event is detected within a boundary area. If the input event is not detected within said boundary area, it is presumed that the input is not caused by two-point user input, and that a single input is performed at the new single input point.

[**0109**] If the input is detected within a boundary area, the second input is detected **88** at the point P_M and the method proceeds as described in **FIG. 8**.

[**0110**] The present method can further comprise steps like determining input areas and calculating boundary areas to speed up the process.

[**0111**] **FIG. 11** depicts schematically a touch based input device controller for a touch based user input device. **FIG. 11** comprises three blocks, a touch based input device **2** such as a touch pad or a touch screen, a touch pad input controller **6**, connected via an interface **4** to said touch pad **2**. The figure further comprises a processor **18** for running an application, which is to be controlled by user input via said touch pad **2**. The controller **6** is connected to the processor **18** via an interface **16**. The controller **6** comprises a memory **8**, a differentiator **10** and first and second evaluation logic **12** and **14**. In the controller **6** the differentiator **10** receives a single position signal from the touch pad **2** and determines the time derivative of the position signal, i.e. the speed at which the signal is moving on said touch pad **2**. The determined value is transferred to the evaluation circuit **12**, to determine if the change of the position signal exceeds a predetermined limit. If the limit is exceeded the signal is regarded as discontinuous, and a dual point user input is identified. The information that a dual-point user input is present is transferred to the second evaluation circuit **14**. The differentiator **10** and the evaluation circuit **12** are provided to determine if dual-point user input occurs or not. If dual-point user input is detected, the second evaluation circuit **14** is used to determine the two actual positions at which a user is expected to touch said touch pad **2**.

[**0112**] The second evaluation circuit **14** uses a formerly stored first position stored in memory **8** and the actual position received via the interface **4** to calculate an actual dual point user input. To calculate both positions of an expected actual dual-point user input, the equations listed in the foregoing specification regarding **FIG. 7** can be used. The second evaluation circuit **14** transfers the calculated dual point user input via the interface **16** to the processor **18** to control an application running on said processor.

[**0113**] The application running on said processor **18** may transfer control information via the interface **16** to the second evaluation circuit **14**.

[**0114**] **FIG. 12** depicts a flowchart of another implementation and embodiment of the method of the present invention. The flowchart comprises substantially three different paths. These paths are described by starting with the shortest path and ending with the longest path. The flowchart starts with a first user input event that is being detected at a position point P_1. It is assumed that the position of the point P_1 is changed and the point is moved. During a detected motion, a signal transition gradient is determined and it is detected if said signal transition gradient exceeds a preset limit. If said signal transition gradient does not exceed said limit, a single input position at the (moved) point P_1 is data reported to an application. This represents the first path through said flowchart.

[**0115**] If said signal transition gradient does exceed said limit a second input event is detected at P_M representing a dual point input, wherein the position P_M represents the center of gravity of a dual point input. In a next step, the two actual input points can be extrapolated from the points P_1 and P_M. In fact, it may be possible to detect the number of contact positions on the touch-device by the resistance (or capacitance) alone. Thus there is no need to detect or compute the positions of the second contact. Though, the second point P_M may not be necessary for the described functionality, it can be used for any kind of application in which the first point is supposed to be moved. Therefore, in a next step the second real input point P_2 is calculated or extrapolated, giving the dual input point data $\{P_1, P_2\}$. On the basis of these data a 'left click event' at P_1 is generated and reported data to an application. This represents the second path in said flowchart.

[**0116**] Before the dual input point data $\{P_1, P_2\}$ are reported to said application, it may happen that a motion of the position of the point P_M is detected. In this case, a signal transition gradient is determined and it is detected if said signal transition gradient exceeds a preset limit. If said signal transition gradient does not exceed said limit, said 'left click event' at P_1 is generated and reported data to an application, as described above.

[**0117**] If said signal transition gradient does exceed said limit, a third input event is detected at P_{MM} that represents a triple point input, wherein the position P_{MM} represents the center of gravity of said triple point input. In fact, it may be possible to detect the number of contact positions on the touch-device by the resistance (or capacitance) alone. Thus, there is no need to detect or compute the positions of the second and third contacts. Though, the third point P_{MM} may